CS202 assignment 2

Tejas R-201050, S Pradeep-200826

February 2022

Problem: Implement a SAT solver. Given a formula in the DIMACS representation, the code returns a model if the formula is satisfiable, else it reports that the formula is unsatisfiable.

We implement a SAT solver using the DPLL algorithm with unit propagation. First we read in the clauses from the input CNF file in the DIMACS representation. We use it to build the clauses in our final formula that needs to be checked for satisfiability, in CNF form. Once we obtain the formula in CNF form, as a list of lists, we use a recursive function called DPLL to simplify the formula until we obtain a model for it. Our backtracking algorithm runs by choosing a literal, assigning a truth value to it, simplifying the formula and then recursively checking if the simplified formula is satisfiable; if this is the case, the original formula is satisfiable; otherwise, the same recursive check is done assuming the opposite truth value. This is known as the splitting rule, as it splits the problem into two simpler sub-problems. The simplification step essentially removes all clauses that become true under the assignment from the formula, and all literals that become false from the remaining clauses.

Unit propagation: If a clause is a unit clause, i.e. it contains only a single unassigned literal, this clause can only be satisfied by assigning the necessary value to make this literal true. Thus, no choice is necessary. Unit propagation consists in removing every clause containing a unit clause's literal and in discarding the complement of a unit clause's literal from every clause containing that complement. In practice, this often leads to deterministic cascades of units, thus avoiding a large part of the naive search space.

We maintain the current list of clauses in a list of lists called cnf, and we maintain the list of literals in cnf in a list cnf_lits. We modify cnf in every recursive call while performing unit propagation. In each recursive call, we assign a truth value to a literal in cnf_lits, simplify the clauses in cnf using unit propagation , and check if cnf is satisfiable. If we find an empty clause then we know that the formula is unsatisfiable, because no interpretation can make the truth value of that clause 1.

Assumptions: We assume that the input is given in the DIMACS format in a .cnf file.

Dependencies: Our code is written in python, and is dependent on the use of the deepcopy library to correctly handle the backtracking algorithm.